GMI Components: Import/Export Variables

Jules Kouatchou

May 12, 2006

Issues Raised at the Meeting

- Include the dimensions of variables

  Done

- Include the units of vraiables

  Done whenever available

- GMAO will provide "press3e" (atmospheric pressure at the edge of each grid box) only and not "press3c", "ai", "bi", "am", "bm" and "pt". These quantities will have to be derived by GMI.

  Though "ai", "bi", "am", "bm" and "pt" may be read in from a MetFields file, there is a GMI routine that sets their values (that remain constant throughout the code). We can then easily derive "press3c". As a matter of fact, we do not even need "press3e". We only need the surface pressure.

- The array "cmi_flags" use for lightning parametrization could be removed for now as IMPORT variable. It is resolution and MetFields dependent.

- Do the variables "con_precip" and "tot_precip" include rainfall or snow?

  Do not know!

- How are "max_cloud" and "ran_cloud" used?

  Inside the Dry Deposition, Emission operators, they are used to compute the fractional cloud cover (local variable):

```
cloud_frac(:,:) =  1.0d0 -  &
            ((1.0d0 - Maxval (max_cloud(:,:,:), dim=3)) *  &
             (Product ((1.0d0 - ran_cloud(:,:,:)), dim=3)))
```

  Inside Gas Phase Chemistry, they are used to derived "tau_cloud" (DAO MetFields only).

```
  Inside the photolysis package, they are use to compute the
  fractional cloud cover.
  Inside the AerosolDust module, they are used to compute the
  optical depth of aerosol/dust species.

- Reduce the dimensions of the emission array, "emiss(:,:,:,:,:)"
  and the species concentartion array, "const(:,:,:,:)".

  A possible solution has been proposed by Tom.

- Check the units of "mcor" and "mass" and they need to be IMPORT
  variables.

  "mcor" and "mass" can internally be derived in the GMI code.
  The GMI unit for "mass" is [kg] but GMAO wants [kg/m2].
  Do not know how to address the issue.

- Why are "emiss_isop", "emiss_monot" and "emiss_nox" considered
  EXPORT variables for the Emission component?

  The three variables are computed by the Emission component and
  passed to the Chemistry component where they are used.

- Is it enough to provide "precipitation" only instead of the variables
  "rain", "rain_zm", "rain_hk" and "rain_ls"?

  They are mainly employed in the wet deposition package.

- What are the arays "s_radius", "s_velocity" and "diffaer"?

  The are updated by the module computing the gravitational settling
  of aerosols and passed to the Dry Deposition operator. We may ignore
  them for this work (till we have a coupled aerosol/combo mechanism).

- What is the array "tropp" and its unit?

  There is a routine in GMI that computes its value using the temperature
  and the atmospheric pressure (at the center of the grid box).
  The unit of this variable is [mb].

- How are the files read by the Emission and Chemistry components?

  Most of the files are read by each worker processor.
  A couple of them are read by the master processor (see below).
```

We list here the varaibles GMI will need from GEOS5 and the ones GMI will pass to GEOS5.

```
=====================================================================
                        EMISSION Component
```

```
======================================================================

IMPORT Variables
----------------

press3c      (:,:,:)  Atmospheric pressure at the center of each grid box [mb    ]
press3e      (:,:,:)  Atmospheric pressure at the edge of each grid box   [mb    ]
zmmu         (:,:,:)  Z-M convective mass flux in     updraft             [Pa/s  ]
cmi_flags    (:,:)    Array of flags that indicate continental, marine, or ice
                      Only used for lightning parameterization.
lwi_flags    (:,:)    Array of flags that indicate land, water, or ice
dtrn         (:,:,:)  Detrainment rate
radswg       (:,:)    Net downward shortwave radiation at ground          [W/m^2 ]
surf_air_temp(:,:)    Surface air temperature                             [degK  ]
surf_rough   (:,:)    Surface roughness                                   [m     ]
con_precip   (:,:)    Convective precipitation                            [mm/day]
tot_precip   (:,:)    Total precipitation                                 [mm/day]
ustar        (:,:)    friction velocity                                   [m/s   ]
max_cloud    (:,:,:)  Maximum overlap cloud fraction for LW
ran_cloud    (:,:,:)  Random  overlap cloud fraction for LW
kel          (:,:,:)  Temperature                                         [degK  ]
pbl          (:,:)    Boundary layer height                               [m     ]
humidity     (:,:,:)  Specific humidity                                   [g/kg  ]
pctm1        (:,:)    Surface pressure at t1                              [mb    ]

emiss    (:,:,:,:,:)  Array of emissions                                   [kg/s  ]

const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]

pt                    pressure = (am * pt) + (bm * psx)                   [mb   ]
ai           (:)      Pressure = (ai * pt) + (bi * psx), ai at zone interface
bi           (:)      Pressure = (ai * pt) + (bi * psx), bi at zone interface
latdeg       (:)      Latitude                                            [deg   ]
londeg       (:)      Longitude                                           [deg   ]
mcor         (:,:)    Area of grid box                                    [m^2   ]
mass         (:,:,:)  Total mass of the atmosphere within each grid box   [kg    ]

EXPORT Variables
----------------
flashrate    (:,:)    Flash rate (flashes per 4x5 box per s)
lightning_no (:,:,:)  3d array of pnox production in kg./sec
emiss_isop   (:,:)    Isoprene    emissions                               [kg/s  ]
emiss_monot  (:,:)    Monoterpene emissions                               [kg/s  ]
emiss_nox    (:,:)    NOx         emissions                               [kg/s  ]
emiss    (:,:,:,:,:)  Array of emissions                                   [kg/s  ]
const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]


======================================================================
                        WET DEPOSITION Component
======================================================================

IMPORT Variables
----------------

press3c      (:,:,:)  Atmospheric pressure at the center of each grid box [mb    ]
press3e      (:,:,:)  Atmospheric pressure at the edge of each grid box   [mb    ]
```

```
kel          (:,:,:)  Temperature                                          [degK  ]
rain         (:,:,:)  rainfall across cell edges                           [mm/day]
rain_zm      (:,:,:)  rain production due to deep conv processes       [kg/kg/sec]
rain_hk      (:,:,:)  rain production due to shal conv processes       [kg/kg/sec]
rain_ls      (:,:,:)  rain production due to lasrge-scale processes    [kg/kg/sec]
coscen       (:)      cosine of latitude of zone centers = cos(dlatr)
grid_height  (:,:,:)  height of each grid box                          [m     ]
mass         (:,:,:)  total mass of the atmosphere within each grid box   [kg    ]
wet_depos    (:,:,:)  wet deposition accumulated since last output        [kg/m^2]
mcor         (:,:)    Area of grid box                                   [m^2   ]
con_precip   (:,:)    Convective precipitation                           [mm/day]
tot_precip   (:,:)    Total precipitation                                [mm/day]
const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]

EXPORT Variables
----------------
wet_depos    (:,:,:)  wet deposition accumulated since last output        [kg/m^2]
const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]
moistq       (:,:,:)  moisture changes due to wet processes             [g/kg/day]


======================================================================
                        DRY DEPOSITION Component
======================================================================

IMPORT Variables
----------------
pt                    pressure = (am * pt) + (bm * psx)                   [mb   ]
ai           (:)      Pressure = (ai * pt) + (bi * psx), ai at zone interface
bi           (:)      Pressure = (ai * pt) + (bi * psx), bi at zone interface
am           (:)      pressure = (am * pt) + (bm * psf), am at zone midpoint
bm           (:)      pressure = (am * pt) + (bm * psf), bm at zone midpoint
latdeg       (:)      Latitude                                           [deg   ]
londeg       (:)      Longitude                                          [deg   ]
mcor         (:,:)    Area of grid box                                   [m^2   ]
mass         (:,:,:)  Total mass of the atmosphere within each grid box   [kg    ]

lwi_flags    (:,:)    array of flags that indicate land, water, or ice
himudity     (:,:,:)  specific humidity                                  [g/kg  ]
max_cloud    (:,:,:)  Maximum overlap cloud fraction for LW
ran_cloud    (:,:,:)  random  overlap cloud fraction for LW
radswg       (:,:)    net downward shortwave radiation at ground         [W/m^2 ]
surf_air_temp(:,:)    surface air temperature                            [degK  ]
surf_rough   (:,:)    surface roughness                                  [m     ]
ustar        (:,:)    friction velocity                                  [m/s   ]
psf          (:,:)    surface pressure field at t1, known at zone centers [mb    ]
kel          (:,:,:)  temperature                                        [degK  ]

s_radius     (:,:,:)  aerosol radius at bottom layer                     [m     ]
s_velocity   (:,:,:)  aerosol settling velocity at bottom layer          [m/s   ]
diffaer      (:,:,:)  aerosol diffusivity at bottom layer                [m^2/s ]

dry_depos    (:,:,:)  dry deposition accumulated since last output        [kg/m^2]
const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]

EXPORT Variables
----------------
```

```
dry_depos    (:,:,:)  dry deposition accumulated since last output        [kg/m^2]
const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]


========================================================================
                        SIMPLE DEPOSITION Component
========================================================================

IMPORT Variables
----------------
press3c      (:,:,:)  atmospheric pressure at the center of each grid box [mb    ]
const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]

EXPORT Variables
----------------
const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]


========================================================================
                        CHEMISTRY Component
========================================================================

IMPORT Variables
----------------
pt                    pressure = (am * pt) + (bm * psx)                    [mb   ]
ai           (:)      Pressure = (ai * pt) + (bi * psx), ai at zone interface
bi           (:)      Pressure = (ai * pt) + (bi * psx), bi at zone interface
am           (:)      pressure = (am * pt) + (bm * psf), am at zone midpoint
bm           (:)      pressure = (am * pt) + (bm * psf), bm at zone midpoint
latdeg       (:)      Latitude                                             [deg   ]
londeg       (:)      Longitude                                            [deg   ]
dlatr        (:)      latitude of zone center in latitude direction        [rad   ]
mcor         (:,:)    Area of grid box                                     [m^2   ]
mass         (:,:,:)  Total mass of the atmosphere within each grid box    [kg    ]
grid_height           height of each grid box                             [m     ]

press3c      (:,:,:)  Atmospheric pressure at the center of each grid box [mb    ]
press3e      (:,:,:)  Atmospheric pressure at the edge of each grid box   [mb    ]
tropp        (:,:)    tropopause pressure                                 [mb    ]
max_cloud    (:,:,:)  Maximum overlap cloud fraction for LW
ran_cloud    (:,:,:)  Random  overlap cloud fraction for LW
tau_cloud    (:,:,:)  optical depth (dimensionless)
kel          (:,:,:)  Temperature                                         [degK  ]
humidity     (:,:,:)  Specific humidity                                   [g/kg  ]
pctm2        (:,:)    Surface pressure at t1+tdt                          [mb    ]
surf_alb_uv  (:,:)    bulk surface albedo (fraction 0-1)
cmf          (:,:,:)  convective mass flux                              [kg/m^2*s]

emiss_isop   (:,:)    Isoprene    emissions                                [kg/s  ]
emiss_monot  (:,:)    Monoterpene emissions                                [kg/s  ]
emiss_nox    (:,:)    NOx         emissions                                [kg/s  ]
emiss    (:,:,:,:,:)  Array of emissions                                   [kg/s  ]

const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]

EXPORT Variables
----------------
```

```
const      (:,:,:,:)  Species concentration, known at zone centers  [mixing ratio]
emiss    (:,:,:,:,:)  Array of emissions                                [kg/s  ]


=========================================================================
                  INPUT FILES NAMES
=========================================================================

    Emission
emiss_infile_name     : emission                input file name
                              read by all the worker processors
light_infile_name     : lightning               input file name
                              read by all the worker processors
precip_infile_name    : precipitation           input file name
                              read by all the worker processors
soil_infile_name      : soil type               input file name
                              read by all the worker processors
isopconv_infile_name  : isoprene convert        input file name
                              read by all the worker processors
monotconv_infile_name : monoterpene convert     input file name
                              read by all the worker processors
veg_infile_name       : vegetation type         input file name
                              read by all the worker processors
lai_infile_name       : leaf area index         input file name
                              read by all the worker processors
gcr_infile_name       : Galactic Cosmic Ray     input file name
                              read by all the worker processors
fertscal_infile_name  : fertilizer scale        input file name
                              read by all the worker processors

    Chemistry
forc_bc_infile_name   : forcing bc              input file name
                              read by master processor
h2oclim_infile_name   : water climatology       input file name
                              read by all the worker processors
lbssad_infile_name    : liq bin sul sad         input file name
                              read by all the worker processors

uvalbedo_infile_name  : uv albedo               input file name
                              read by master processor
cross_section_file    : X-Section quantum yield input file name
                              read by all the worker processors
rate_file             : Master rate             input file name
                              read by all the worker processors
T_O3_climatology_file : T & O3 climatology      input file name
                              read by all the worker processors

=========================================================================
      s_radius, s_velocity and diffaer
=========================================================================
s_radius, s_velocity and diffaer are updated inside the routine
computing the gravitational settling of aerosols, Update_Grav_Settling.
Update_Grav_Settling is called only if the logical variable "do_grav_set"
is set to true.
```

How some Variables are Used in the GMI Code

```
======================================================================
    mcor, pt, ai, bi, am, bm
======================================================================

==================
EMISSION Component
==================
"mcor" is employed to
       - update the array "emiss_nox" (unit issue?)
       - update diagnostics variables

"humidity, ai, bi, pt" are only used to compute the grid box height
inside Add_Emiss_Llnl. They can be removed from the argument list of
the emission control routine and be replaced by "grid_height".

==================
CHEMISTRY Component
==================
"mcor" is utilized
       - to calculate column ozone in the lookup table module (unit issue?)
       - for surface emission diagnostics.

"mcor, ai, bi" are used to calculate conversion to go from kg/box/s to
mole/cm^3/s.

"humidity, ai, bi, pt" are only used to compute the grid box height
inside Update_Semiss_Inchem. They can be removed from the argument list of
this routine and be replaced by "grid_height".
The same can be said for the argument list of
     - Update_QuadChem
     - Update_Smv2chem

"ai, bi, pt" are employed to compute tau_cloud for DAO Met Fields.

"ai, bi, am, bm, pt" are used in the photolysis package to calculate
pressure at boundaries of CTM levels:

   Press(:) = ai(:)*pt + bi(:)*SurfPressure
   Press(:) = am(:)*pt + bm(:)*SurfPressure

===================
DEPOSITION Component
===================
"mcor" is utilized to
       - update dry_depos and wet_depos (unit issue?)
       - update moistq and precip_bot (unit issue?)

"humidity, ai, bi, pt" are only used to compute the grid box height
inside Update_Drydep. They can be removed from the argument list of
this routine and be replaced by "grid_height".
"humidity, am, bm, pt" are also used there to compute something
similar to the grid box height.

===================
CONVECTION Component
```

```
====================
"mcor" is utilized to
       - compute the internal number of time step for convection
       - update the wet_depos array (unit issue?)

"ai, bi, pt" only employed in Do_Convec_Dao2.

====================
ADVECTION Component
====================
"mcor, ai, bi, pt" are used the total mass for internal diagnostics.
The variable "mass" can passed in the argument list of the advection
control routine and replace them.
```